

WRDC-TR-90-8007
Volume VII
Part 6

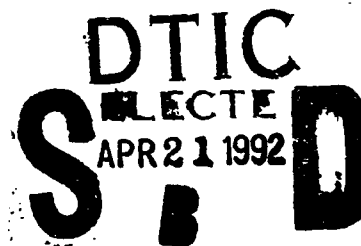
AD-A248 915



INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume VII - Communications Subsystem
Part 6 - File Input/Output Primitives (FIOPS) Unit Test Plan

S. Barker

Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209



September 1990

Final Report for Period 1 April 1987 - 31 December 1990

Approved for Public Release; Distribution is Unlimited

MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533

92 4 20 008

92-09981

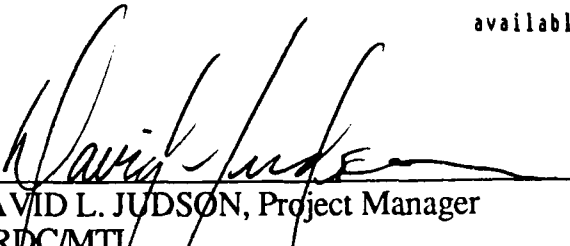


NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

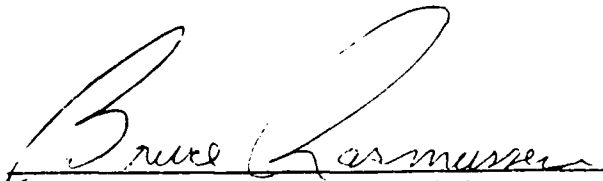
This technical report has been reviewed and is approved for publication.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations


DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

FOR THE COMMANDER:


BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS None	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UTP620343400		5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR-90-8007 Vol. VII, Part 6	
6a. NAME OF PERFORMING ORGANIZATION Control Data Corporation; Integration Technology Services	6b. OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION WRDC/MTI	
6c. ADDRESS (City, State, and ZIP Code) 2970 Presidential Drive Fairborn, OH 45324-6209		7b. ADDRESS (City, State, and ZIP Code) WPAFB, OH 45433-6533	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Wright Research and Development Center, Air Force Systems Command, USAF	8b. OFFICE SYMBOL (if applicable) WRDC/MTI	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM. F33600-87-C-0464	
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, Ohio 45433-6533		10. SOURCE OF FUNDING NOS.	
11. TITLE File I/ See block 19		PROGRAM ELEMENT NO. 78011F	PROJECT NO. 595600
		TASK NO. F95600	WORK UNIT NO. 20950607
12. PERSONAL AUTHOR(S) Structural Dynamics Research Corporation: Barker, S., et al.			
13a. TYPE OF REPORT Final Report	13b. TIME COVERED 4/1/87-12/31/90	14. DATE OF REPORT (Yr., Mo., Day) 1990 September 30	15. PAGE COUNT 32
16. SUPPLEMENTARY NOTATION WRDC/MTI Project Priority 6203			
17. COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify block no.)	
FIELD	GROUP	SUB GR.	
1308	0905		
19. ABSTRACT (Continue on reverse if necessary and identify block number)			
<p>This unit test plan establishes the methodology and procedures used to adequately test the capabilities of the computer program identified as I/O Primitives (FIOPS).</p> <p>BLOCK 11:</p> <p>INTEGRATED INFORMATION SUPPORT SYSTEM Vol VII - Communications Subsystem</p> <p>Part 6 - File Input/Output Primitives (FIOPS) Unit Test Plan</p>			
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS		21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL David L. Judson		22b. TELEPHONE NO. (Include Area Code) (513) 255-7371	22c. OFFICE SYMBOL WRDC/MTI

FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

<u>SUBCONTRACTOR</u>	<u>ROLE</u>
Control Data Corporation	Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.
D. Appleton Company	Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.
ONTEK	Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.
Simpact Corporation	Responsible for Communication development.
Structural Dynamics Research Corporation	Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.
Arizona State University	Responsible for test bed operations and support.

TABLE OF CONTENTS

		<u>Page</u>
SECTION 1.0	GENERAL	1-1
1.1	Purpose	1-1
1.2	Project References	1-1
1.3	Terms and Abbreviations	1-1
SECTION 2.0	DEVELOPMENT ACTIVITY	2-1
2.1	Statement of Pretest Activity	2-1
2.2	Pretest Activity Results	2-1
SECTION 3.0	SYSTEM DESCRIPTION	3-1
3.1	System Description	3-1
3.2	Testing Schedule	3-1
3.3	First Location Testing	3-1
3.4	Subsequent Location Testing	3-1
SECTION 4.0	SPECIFICATIONS AND EVALUATIONS	4-1
4.1	Test Specifications	4-1
4.2	Testing Methods and Constraints	4-1
4.3	Test Progression	4-1
4.4	Test Evaluation	4-2
4.4.1	VAX Test Evaluation	4-2
4.4.2	IBM Test Evaluation	4-2
4.5	Error Handling	4-2
SECTION 5.0	TEST PROCEDURES	5-1
5.1	Test Description	5-1
5.2	Test Control	5-1
5.3	Test Procedures	5-1
5.3.1	VAX Test Procedures	5-1
5.3.2	IBM Test Procedures	5-3

APPENDICES

A	FILE1T INPUT FILE 1	A-1
B	FILE2T INPUT FILE 2	B-1
C	FILET OUTPUT FILE	C-1
D	RUN TIME OUTPUT TO THE SCREEN ON THE VAX	D-1
E	STATUS ERROR CODES	E-1
F	FILE1T.DAT NOT FOUND ERROR	F-1
G	FILE2T.DAT NOT FOUND ERROR	G-1
H	FILE2T.DAT RECORDS NOT FIXED LENGTH ERROR	H-1
I	FILE2T.DAT OUTPUT FILE 2	I-1

LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Title</u>	<u>Page</u>
5-1	FILE1T.DAT RECORDS	5-2
5-2	FILE1T INPUT FILE RECORDS	5-3

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

SECTION 1

GENERAL

1.1 Purpose

This unit test plan establishes the methodology and procedures used to adequately test the capabilities of the computer programs identified as the File I/O Primitives (FIOPs). The FIOP's are one configuration item of the IISS Interprocess Communications Subsystem.

1.2 Project References

- [1] ICAM Documentation Standards, 15 September 1983, IDS150120000C.
- [2] Project Master Plan / Schedule: ICAM IDBM Project 6202, 6 March, PMP620220000.
- [3] IISS System Requirements Document, 1 November 1985, SRD620140000.
- [4] IISS System Design Specification, 1 November 1985, SDS620140000A.
- [5] IISS SDM Subsystem Development Specification: Distributed Request Supervisor, 1 November 1985, DS 620141310.
- [6] IISS CDM Subsystem Development Specification: aggregator, 1 November 1985, DS 620141320.
- [7] IISS SDM Subsystem Development Specification: File Utilities, 1 November 1985, DS 620141330.
- [8] IISS Configuration Management: SCM Administrator's Manual, 1 November 1985, CMA620124000.

1.3 Terms and Abbreviations

AP: Term used for application programs within the IISS.

Bugs: Errors that result during the execution of a code.

Digital Equipment Corporation (DEC): The minicomputer company which manufactures, among others, the VAX series of computers and micro machines and large scale computers and systems.

International Business Machines Corporation (IBM): The company that manufactures the test bed IBM 4381.

Integrated Information Support System (IISS): A product of the ICAM Integrated Data Base Management development program which is designed to connect heterogeneous computers and provide for distributed real-time data base access.

Multiple Virtual Storage/eXtended Architecture (MVS/XA): An operating system that executes on large IBM computers.

Scripting: The process of running a program and relaying input and output data in such a manner as to describe the run time test activity of a program.

Virtual Address Extension (VAX): A type of minicomputer manufactured by DEC which is used at the test bed.

Virtual Memory System (VMS): An Operating System that executes on DEC VAX computers and is used on the test bed VAX.

SECTION 2

DEVELOPMENT ACTIVITY

2.1 Statement of Pretest Activity

During system development, the computer programs were tested progressively. Functionality was incrementally tested and as bugs were discovered by this testing, the software was corrected.

This testing was conducted by the individual program developer in a manual mode. Any errors were noted by the developer and corrections to the program were made after a testing session.

2.2 Pretest Activity Results

Testing of the programs discovered a few minor errors which were then corrected and retesting proved to be successful. Testing included exceptional conditions and error conditions for the language. The overall test results during development showed no major programming errors. Only minor bugs were discovered and corrected.

SECTION 3

SYSTEM DESCRIPTION

3.1 System Description

This program provides an interface independent of the operating system. The user will be able to access files on this independent neutral system without knowledge of system specific information.

There are seven File I/O Primitive (FIOP) routines: "NAMFIL", "OPNFIL", "INPFIL", "OUTFIL", "SEKFIL", "CLSFIL", and "SRTFIL". The FIOPs will use the preexisting system dependent routines that exist on the VAX and the IBM computers for accessing files and sorting the data in them. The FIOPs basically conform to the normal file operations described in any programming language, but have two additions. The Temporary File Name function and the File Sort/Merge are actually file utilities.

3.2 Testing Schedule

The execution of this program is dependent on the Air Force test bed being up and running on the VAX and the IBM.

3.3 First Location Testing

These tests require the following:

Equipment: IISS Air Force test bed VAX and IBM.

Support Software: VAX "C", IBM Assembler

Deliverables: File I/O Primitives Source Code.

Test Materials: the test program, FIOPTST on the VAX and FIOPTST on the IBM, and two input files, FILE1T.DAT and FILE2T.DAT on the VAX and IISSCM.UI.R23.FILE1T and IISSCM.UI.R23.FILE2T on the IBM.

Security Considerations: None

Personnel: Integrator familiar with IISS on VAX and IBM.

Training: IISS Users Guide

3.4 Subsequent Location Testing

The requirements as listed above need to be met. Please reference the VAX specific input and output files listed in Section 5.3.1 and IBM specific files listed in Section 5.3.2.

SECTION 4

SPECIFICATIONS AND EVALUATIONS

4.1 Test Specifications

The following requirements are demonstrated by the outlined tests:

Functional Requirements	Test Activity	
	A	B
Open File	*	*
Input File	*	*
Temporary File Name	*	*
Close File	*	*
Output File	*	*
Seek Forward/Backward One Record	*	*
Sort/Merge File	*	*

A Run the File I/O Primitives program on the Test Bed VAX.

B Run the File I/O Primitives program on the IBM Test Bed.

The steps outlined in section 5-3 and the files in Appendices A, B, C, D, E, I and J show the direct correspondence between the test and the functional requirements as listed in this section.

4.2 Testing Methods and Constraints

The tests as outlined in Section 5.3 must be followed. The required input is stated for each test. This procedure tests the normal mode of operation of these functions. These tests have been done, however, through the normal testing done by the developer of these functions. No data recording is required and no additional constraints are placed on this unit test besides those listed in Section 5.3. In order to run this test two input files are necessary; as well as the test programs, FIOPTST on VAX and FIOPTST on IBM.

4.3 Test Progression

The progression of testing of the File I/O Primitives is fully outlined in Section 5.3 of this unit test plan. This progression should be followed exactly to insure the successful testing of this IISS configuration item.

4.4 Test Evaluation

There is one stage in the testing of the File I/O Primitives.

Stage 1: Run the test program with the two input files.

4.4.1 VAX Test Evaluation

The test results are evaluated by observing that the run time output to the screen matches the listing in Appendix D. The generated output files, FILET.OUTPUT and FILE2T.OUTPUT listed must also match the listing in Appendices C and I respectively.

4.4.2 IBM Test Evaluation

The test results are evaluated by observing that the run time output to the screen is similar to the listing in Appendix D. (NOTE that this listing is VAX generated therefore the filenames will be different on the IBM. The generated output files, IISSCM.R23.UI.FILET and IISSCM.R23.UI.FILE2TO must also match the files listed in Appendices C and I respectively.

4.5 Error Handling

Error Handling test cases were developed for the File I/O Primitives. These test cases are described in the following matrix. Error results from these tests are documented in Appendices F, G, and H. For further reference, the Status/Error Codes are listed in Appendix E.

Error Handling Tests	Test Activity C D
Copy File1 into File4 with an incorrect input file in File1	* *
Copy File2 into File5 with an incorrect input file in File2 (see Appendix H for results)	* *
Attempt a run of FIOPTST without the input file FILE1T.DAT (see Appendix F for results)	* *
Attempt a run of FIOPTST without the input file FILE2T.DAT (see Appendix G for results)	* *
C Error causing runs of the File I/O Primitives program on the Test Bed VAX.	
D Error causing runs of the File I/O Primitives program on the IBM Test Bed.	

SECTION 5

TEST PROCEDURES

5.1 Test Description

This test uses a test program, FIOPTST on the VAX and FIOPTST on the IBM, and two input data files to test the File I/O Primitives. The test cases are explicit to the host computer. The testing of the File I/O Primitives functional requirements is embodied in test cases A and B. Error handling testing is accomplished using test cases C and D.

- A Run the File I/O Primitives program on the Test Bed VAX.
- B Run the File I/O Primitives program on the IBM Test Bed.
- C Error causing runs of the File I/O Primitives program on the Test Bed VAX.
- D Error causing runs of the File I/O Primitives program on the IBM Test Bed.

5.2 Test Control

As outlined, this unit test is a manual test which may be done by anyone. The required input data is documented for each function being tested and the resulting successful output is also documented. The order of the testing is also completely documented. The test control information is described in Section 5.3. Verification of the test is by a manual comparison of the test output with the expected results as they are documented here.

5.3 Test Procedures

The File I/O Primitives program will transcend host computers. The host computers will be the VAX and the IBM. The following test procedures explain both VAX and IBM test procedures.

5.3.1 VAX Test Procedures

To run the File I/O Primitives program you must be logged onto the VAX. You must have the source code FIOPTST.COB. You must also have the two necessary input files used by the program, FILE1T.DAT and FILE2T.DAT. NOTE that these files are stored in IISS Configuration Management with a .FLR extension and must be renamed before executing the test. The user must have a copy of the program in his directory along with the two input files FILE1T.DAT and FILE2T.DAT. A listing of these input files can be found in Appendices A and B. When the user runs the program the run time output to the screen listed in Appendix D will result. Two output files (FILET.OUTPUT and FILE2T.OUTPUT) will also be generated. These files are listed in Appendices C and I respectively.

Error flags exist at each status point in the run time output seen in Appendix D. A listing of status code errors and messages can be found in Appendix E. If an error occurs at any point there will be a status number indicating the error that has occurred and the program will stop and return a prompt. For example, if an improper input file were to be used the following output would occur as shown in Figure 5-1.

```
11102 OPNFIL Records are not Fixed_Length
OPEN FILE FILE1T.DAT
MODE = READ
STAT = 11102
LAST STATUS = 11102
```

Figure 5-1. FILE1T.DAT RECORDS

Status codes are defined as strings (arrays) of 5 characters. A return code of all zeros ("00000") indicates successful completion of the called function, otherwise an error has occurred. The error codes returned are formatted in such a way that one can distinguish the subsystem function being performed, and have an idea about what kind of error occurred. The FIOPs calls ERRPRO with an accompanying descriptive message to be put in the ERRLOG file if the error is fatal. If the error code is a warning, ERRPRO will not be called. The left most two characters are "11". The left "1" indicates "primitives", and the right "1" indicates "File I/O" Primitives. The middle character refers to the FIOP function performed, and the right-most two digits refers to function specific errors. Further, the right-most digit indicates the severity of the error, with a zero indicating a warning, and a nonzero indicating fatal errors.

Not all status codes are errors. For instance 11130 is a status generated during this particular run and listed in Appendix D. The 11130 status is CREATED NEW FILE FOR APPEND.

As the program executes, three temporary (.TMP) files are created in the directory. These files are work files which contain certain information generated during run time. The .TMP files represent the following functions:

1. Copy FILE2 to FILE5, one .TMP file will contain FILE5.
2. Copy FILE1 and FILE2 into FILE3 Alternating Recs; FILE3 is then copied to FILE6; and one .TMP file will contain FILE6.
3. One .TMP file will contain a sorted output of FILE3.

The generated output file FILE2T.OUTPUT is FILE3 when FILE1 and FILE2 are copied into FILE3 by alternating records.

5.3.2 IBM Test Procedures

The FIOPTTEST program must be linked using the LKFIOP JCL. The output files IISSCM.R23.FILET and IISSCM.R23.FILETO must also be allocated as sequential files with the following DCB parameters, RECFM=FB,LRECL=80,BLKSIZE=800.

When testing on the IBM host, the test team logs onto Time Sharing Options (TSO) with a session that combines the running of IISS APs with the operation of the IISS system itself. The CLIST SIGNON is executed as follows to allocate the IISS datasets:

```
EXEC 'IISSCM.CLIST(SIGNON)'
```

The program OSIMVSI is executed as follows to start IISS:

```
CALL 'IISSCM.R23.LOADLIB(OSIMVSI)' 'UTEST'
```

Enter FIOPTTEST in response to the "ENTER PROGRAM NAME TO EXECUTE OR END TO EXIT" prompt.

This program reads the input files IISSCM.R23.UI.FILE1T and IISSCM.R23.UI.FILE2T. A listing of these input files can be found in Appendices A and B. When the user runs the program the run time output to the screen, as listed in Appendix D, results. The data listed in Appendices C and I will also be written to the files IISSCM.R23.FILET and IISSCM.R23.FILETO, respectively.

Error flags exist at each status point in the run time output seen in Appendix D. A listing of status code errors and messages can be found in Appendix E. If an error occurs at any point there will be a status number indicating the error that has occurred and the program will stop and return a prompt. For example, if an improper input file were to be used the following output would occur as shown in Figure 5-2.

```
11102 OPNFIL Records are not Fixed_Length
OPEN FILE FILE1T.DAT
MODE = READ
STAT = 11102
LAST STATUS = 11102
```

Figure 5-2. FILE1T INPUT FILE RECORDS

Status codes are defined as strings (arrays) of 5 characters. A return code of all zeros ("00000") indicates successful completion of the called function, otherwise an error has occurred. The error codes returned are formatted in such a way that one can distinguish the subsystem function being performed, and have an idea about what kind of error occurred. The FIOPs will call ERRPRO with an accompanying descriptive message to be put in the ERRLOG file if the error is fatal. If the error code is a warning, ERRPRO will not be called. The left most two characters are "11". The left "1" indicates "primitives", and the right "1" indicates "File I/O" Primitives. The middle character refers to the FIOp function performed, and

the right-most two digits will refer to function specific errors. Further, the right-most digit will indicate the severity of the error, a zero will be for warnings, and a nonzero will be for fatal errors.

Not all status codes are errors. For instance 11130 is a status generated during this particular run and listed in Appendix D. The 11130 status is CREATED NEW FILE FOR APPEND.

As the program executes, three temporary files will be created. These files are work files which contain certain information generated during run time. These files represent the following functions.

1. Copy FILE2 to FILE5, one temporary file will contain FILE5.
2. Copy FILE1 and FILE2 into FILE3 Alternating Recs; FILE3 is then copied to FILE6; and one temporary file will contain FILE6.
3. One temporary file will contain a sorted output of FILE3.

The data written to IISSCM.R23.FILE2TO represents FILE3 when FILE1 and FILE2 are copied into FILE3 by alternating records.

APPENDIX A

FILE1T INPUT FILE 1

FILE1T.DAT ON THE VAX; IISSCM.R30.UI.FILE1T ON THE IBM

FILE1	RECORD01	TRUNCATED
FILE1	RECORD02	TRUNCATED
FILE1	RECORD03	TRUNCATED
FILE1	RECORD04	TRUNCATED
FILE1	RECORD05	TRUNCATED
FILE1	RECORD06	TRUNCATED
FILE1	RECORD07	TRUNCATED
FILE1	RECORD08	TRUNCATED
FILE1	RECORD09	TRUNCATED
FILE1	RECORD10	TRUNCATED
FILE1	RECORD11	TRUNCATED
FILE1	RECORD12	TRUNCATED
FILE1	RECORD13	TRUNCATED
FILE1	RECORD14	TRUNCATED
FILE1	RECORD15	TRUNCATED
FILE1	RECORD16	TRUNCATED
FILE1	RECORD17	TRUNCATED
FILE1	RECORD18	TRUNCATED
FILE1	RECORD19	TRUNCATED
FILE1	RECORD20	TRUNCATED
FILE1	RECORD21	TRUNCATED
FILE1	RECORD22	TRUNCATED
FILE1	RECORD23	TRUNCATED
FILE1	RECORD24	TRUNCATED
FILE1	RECORD25	TRUNCATED
FILE1	RECORD26	TRUNCATED
FILE1	RECORD27	TRUNCATED
FILE1	RECORD28	TRUNCATED
FILE1	RECORD29	TRUNCATED
FILE1	RECORD30	TRUNCATED
FILE1	RECORD31	TRUNCATED
FILE1	RECORD32	TRUNCATED
FILE1	RECORD33	TRUNCATED
FILE1	RECORD34	TRUNCATED
FILE1	RECORD35	TRUNCATED
FILE1	RECORD36	TRUNCATED
FILE1	RECORD37	TRUNCATED
FILE1	RECORD38	TRUNCATED
FILE1	RECORD39	TRUNCATED
FILE1	RECORD40	TRUNCATED
FILE1	RECORD41	TRUNCATED
FILE1	RECORD42	TRUNCATED
FILE1	RECORD43	TRUNCATED
FILE1	RECORD44	TRUNCATED
FILE1	RECORD45	TRUNCATED
FILE1	RECORD46	TRUNCATED
FILE1	RECORD47	TRUNCATED
FILE1	RECORD48	TRUNCATED
FILE1	RECORD49	TRUNCATED

APPENDIX B

FILE2T INPUT FILE 2

FILE2T.DAT ON THE VAX; IISSCM.R23.UI.FILE2T ON THE IBM

FILE2 RECORD01
FILE2 RECORD02
FILE2 RECORD03
FILE2 RECORD04
FILE2 RECORD05
FILE2 RECORD06
FILE2 RECORD07
FILE2 RECORD08
FILE2 RECORD09
FILE2 RECORD10
FILE2 RECORD11
FILE2 RECORD12
FILE2 RECORD13
FILE2 RECORD14
FILE2 RECORD15
FILE2 RECORD16
FILE2 RECORD17
FILE2 RECORD18
FILE2 RECORD19
FIEL2 RECORD20

APPENDIX C

FILET OUTPUT FILE

FILET.OUTPUT ON THE VAX; IISSCM.R23.FILET ON THE IBM

FILE1	RECORD01	TRUNC
FILE2	RECORD01	
FILE1	RECORD02	TRUNC
FILE2	RECORD02	
FILE1	RECORD03	TRUNC
FILE2	RECORD03	
FILE1	RECORD04	TRUNC
FILE2	RECORD04	
FILE1	RECORD05	TRUNC
FILE2	RECORD05	
FILE1	RECORD06	TRUNC
FILE2	RECORD06	
FILE1	RECORD07	TRUNC
FILE2	RECORD07	
FILE1	RECORD08	TRUNC
FILE2	RECORD08	
FILE1	RECORD09	TRUNC
FILE2	RECORD09	
FILE1	RECORD10	TRUNC
FILE2	RECORD10	
FILE1	RECORD11	TRUNC
FILE2	RECORD11	
FILE1	RECORD12	TRUNC
FILE2	RECORD12	
FILE1	RECORD13	TRUNC
FILE2	RECORD14	
FILE1	RECORD14	TRUNC
FILE2	RECORD15	
FILE1	RECORD16	TRUNC
FILE2	RECORD16	
FILE1	RECORD17	TRUNC
FILE2	RECORD17	
FILE1	RECORD18	TRUNC
FILE2	RECORD18	
FILE1	RECORD19	TRUNC
FILE2	RECORD19	
FILE1	RECORD20	TRUNC
FILE2	RECORD20	
FILE1	RECORD21	TRUNC
FILE1	RECORD22	TRUNC
FILE1	RECORD23	TRUNC
FILE1	RECORD24	TRUNC
FILE1	RECORD25	TRUNC
FILE1	RECORD26	TRUNC
FILE1	RECORD27	TRUNC
FILE1	RECORD28	TRUNC
FILE1	RECORD29	TRUNC
FILE1	RECORD30	TRUNC
FILE1	RECORD31	TRUNC
FILE1	RECORD32	TRUNC
FILE1	RECORD34	TRUNC

FILE1 RECORD35
FILE1 RECORD36
FILE1 RECORD37
FILE1 RECORD38
FILE1 RECORD39
FILE1 RECORD40
FILE1 RECORD41
FILE1 RECORD42
FILE1 RECORD43
FILE1 RECORD44
FILE1 RECORD45
FILE1 RECORD46
FILE1 RECORD47
FILE1 RECORD48
FILE1 RECORD49

TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC
TRUNC

APPENDIX D

RUN TIME OUTPUT TO THE SCREEN ON THE VAX

```
OPEN FILE FILE1T.DAT
  MODE = READ
  STAT = 00000
OPEN FILE FILE2T.DAT
  MODE = READ
  STAT = 00000
OPEN FILE FILET.OUTPUT
  MODE = WRITE
  STAT = 00000
FILENAME = IISS_SDR:[DIR]FS_00002C3D_87320_1134224_01.TMP
OPEN FILE IISS_SDR:[DIR]FS_00002C3D_87320_11342724_01.TMP
  MODE = WRITE
  STAT = 00000
FILENAME = IISS_SDR:[DIR]FS_00002C3D_87320_11342745_02.TMP
OPEN FILE IISS_SDR:[DIR]FS_00002C3D_87320_11342745_02.TMP
  MODE = WRITE
  STAT = 00000
RECORD 1 = FILE1  RECORD01                                TRUNCATED
RECORD 2 = FILE2  RECORD01
RECORD 1 = FILE1  RECORD02                                TRUNCATED
RECORD 2 = FILE2  RECORD02
RECORD 1 = FILE1  RECORD03                                TRUNCATED
RECORD 2 = FILE2  RECORD03
RECORD 1 = FILE1  RECORD04                                TRUNCATED
RECORD 2 = FILE2  RECORD04
RECORD 1 = FILE1  RECORD05                                TRUNCATED
RECORD 2 = FILE2  RECORD05
RECORD 1 = FILE1  RECORD06                                TRUNCATED
RECORD 2 = FILE2  RECORD06
RECORD 1 = FILE1  RECORD07                                TRUNCATED
RECORD 2 = FILE2  RECORD07
RECORD 1 = FILE1  RECORD08                                TRUNCATED
RECORD 2 = FILE2  RECORD08
RECORD 1 = FILE1  RECORD09                                TRUNCATED
RECORD 2 = FILE2  RECORD09
RECORD 1 = FILE1  RECORD10                                TRUNCATED
RECORD 2 = FILE2  RECORD10
RECORD 1 = FILE1  RECORD11                                TRUNCATED
RECORD 2 = FILE2  RECORD11
RECORD 1 = FILE1  RECORD12                                TRUNCATED
RECORD 2 = FILE2  RECORD12
RECORD 1 = FILE1  RECORD13                                TRUNCATED
RECORD 2 = FILE2  RECORD13
RECORD 1 = FILE1  RECORD14                                TRUNCATED
RECORD 2 = FILE2  RECORD14
RECORD 1 = FILE1  RECORD15                                TRUNCATED
RECORD 2 = FILE2  RECORD15
RECORD 1 = FILE1  RECORD16                                TRUNCATED
RECORD 2 = FILE2  RECORD16
RECORD 1 = FILE1  RECORD17                                TRUNCATED
RECORD 2 = FILE2  RECORD17
RECORD 1 = FILE1  RECORD18                                TRUNCATED
RECORD 2 = FILE2  RECORD18
```

```

RECORD 1 = FILE1 RECORD19 TRUNCATED
RECORD 2 = FILE2 RECORD19
RECORD 1 = FILE1 RECORD20 TRUNCATED
RECORD 2 = FILE2 RECORD20
RECORD 1 = FILE1 RECORD21 TRUNCATED
RECORD 1 = FILE1 RECORD22 TRUNCATED
RECORD 1 = FILE1 RECORD23 TRUNCATED
RECORD 1 = FILE1 RECORD24 TRUNCATED
RECORD 1 = FILE1 RECORD25 TRUNCATED
RECORD 1 = FILE1 RECORD26 TRUNCATED
RECORD 1 = FILE1 RECORD27 TRUNCATED
RECORD 1 = FILE1 RECORD28 TRUNCATED
RECORD 1 = FILE1 RECORD29 TRUNCATED
RECORD 1 = FILE1 RECORD30 TRUNCATED
RECORD 1 = FILE1 RECORD31 TRUNCATED
RECORD 1 = FILE1 RECORD32 TRUNCATED
RECORD 1 = FILE1 RECORD33 TRUNCATED
RECORD 1 = FILE1 RECORD34 TRUNCATED
RECORD 1 = FILE1 RECORD35 TRUNCATED
RECORD 1 = FILE1 RECORD36 TRUNCATED
RECORD 1 = FILE1 RECORD37 TRUNCATED
RECORD 1 = FILE1 RECORD38 TRUNCATED
RECORD 1 = FILE1 RECORD39 TRUNCATED
RECORD 1 = FILE1 RECORD40 TRUNCATED
RECORD 1 = FILE1 RECORD41 TRUNCATED
RECORD 1 = FILE1 RECORD42 TRUNCATED
RECORD 1 = FILE1 RECORD43 TRUNCATED
RECORD 1 = FILE1 RECORD44 TRUNCATED
RECORD 1 = FILE1 RECORD45 TRUNCATED
RECORD 1 = FILE1 RECORD46 TRUNCATED
RECORD 1 = FILE1 RECORD47 TRUNCATED
RECORD 1 = FILE1 RECORD48 TRUNCATED
RECORD 1 = FILE1 RECORD49 TRUNCATED
CLOSING FILE FILE1T.DAT
CLOSING FILE FILE2T.DAT
CLOSING FILE FILET.OUTPUT
CLOSING FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11342724_01.TMP
CLOSING FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11342745_02.TMP
OPEN FILE FILET.OUTPUT
MODE = READ
STAT = 00000
FILENAME = IISS_SDRG:[DIR]FS_00002C3D_87320_11343062_03.TMP
OPEN FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11343062_03.TMP
MODE = APPEND
STAT = 11130
IISS_SDRG:
[DIR]FS_00002C3D_87320_11343062_03.TMP DID NOT EXIST;CREATE
RECORD 3 = FILE1 RECORD01 TRUNC
RECORD 3 = FILE2 RECORD01
RECORD 3 = FILE1 RECORD02 TRUNC
RECORD 3 = FILE2 RECORD02
RECORD 3 = FILE1 RECORD03 TRUNC
RECORD 3 = FILE2 RECORD03
RECORD 3 = FILE1 RECORD04 TRUNC
RECORD 3 = FILE2 RECORD04
RECORD 3 = FILE1 RECORD05 TRUNC
RECORD 3 = FILE2 RECORD05

```

RECORD 3 = FILE1	RECORD06	TRUNC
RECORD 3 = FILE2	RECORD06	
RECORD 3 = FILE1	RECORD07	TRUNC
RECORD 3 = FILE2	RECORD07	
RECORD 3 = FILE1	RECORD08	TRUNC
RECORD 3 = FILE2	RECORD08	
RECORD 3 = FILE1	RECORD09	TRUNC
RECORD 3 = FILE2	RECORD09	
RECORD 3 = FILE1	RECORD10	TRUNC
RECORD 3 = FILE2	RECORD10	
RECORD 3 = FILE1	RECORD11	TRUNC
RECORD 3 = FILE2	RECORD11	
RECORD 3 = FILE1	RECORD12	TRUNC
RECORD 3 = FILE2	RECORD12	
RECORD 3 = FILE1	RECORD13	TRUNC
RECORD 3 = FILE2	RECORD13	
RECORD 3 = FILE1	RECORD14	TRUNC
RECORD 3 = FILE2	RECORD14	
RECORD 3 = FILE1	RECORD15	TRUNC
RECORD 3 = FILE2	RECORD15	
RECORD 3 = FILE1	RECORD16	TRUNC
RECORD 3 = FILE2	RECORD16	
RECORD 3 = FILE1	RECORD17	TRUNC
RECORD 3 = FILE2	RECORD17	
RECORD 3 = FILE1	RECORD18	TRUNC
RECORD 3 = FILE2	RECORD18	
RECORD 3 = FILE1	RECORD19	TRUNC
RECORD 3 = FILE2	RECORD19	
RECORD 3 = FILE1	RECORD20	TRUNC
RECORD 3 = FILE2	RECORD20	
RECORD 3 = FILE1	RECORD21	TRUNC
RECORD 3 = FILE1	RECORD22	TRUNC
RECORD 3 = FILE1	RECORD23	TRUNC
RECORD 3 = FILE1	RECORD24	TRUNC
RECORD 3 = FILE1	RECORD25	TRUNC
RECORD 3 = FILE1	RECORD26	TRUNC
RECORD 3 = FILE1	RECORD27	TRUNC
RECORD 3 = FILE1	RECORD28	TRUNC
RECORD 3 = FILE1	RECORD29	TRUNC
RECORD 3 = FILE1	RECORD30	TRUNC
RECORD 3 = FILE1	RECORD31	TRUNC
RECORD 3 = FILE1	RECORD32	TRUNC
RECORD 3 = FILE1	RECORD33	TRUNC
RECORD 3 = FILE1	RECORD34	TRUNC
RECORD 3 = FILE1	RECORD35	TRUNC
RECORD 3 = FILE1	RECORD36	TRUNC
RECORD 3 = FILE1	RECORD37	TRUNC
RECORD 3 = FILE1	RECORD38	TRUNC
RECORD 3 = FILE1	RECORD39	TRUNC
RECORD 3 = FILE1	RECORD40	TRUNC
RECORD 3 = FILE1	RECORD41	TRUNC
RECORD 3 = FILE1	RECORD42	TRUNC
RECORD 3 = FILE1	RECORD43	TRUNC
RECORD 3 = FILE1	RECORD44	TRUNC
RECORD 3 = FILE1	RECORD45	TRUNC
RECORD 3 = FILE1	RECORD46	TRUNC
RECORD 3 = FILE1	RECORD47	TRUNC

```
RECORD 3 = FILE1 RECORD48 TRUNC
RECORD 3 = FILE1 RECORD49 TRUNC
CLOSING FILE FILET.OUTPUT
CLOSING FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11343062_03.TMP
OPEN FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11342724_01.TMP
MODE = READ
STAT = 00000
OPEN FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11343062_03.TMP
MODE = APPEND
STAT = 00000
RECORD 4 = FILE1 RECORD01 TRUNC
RECORD 4 = FILE1 RECORD03 TRUNC
RECORD 4 = FILE1 RECORD05 TRUNC
RECORD 4 = FILE1 RECORD07 TRUNC
RECORD 4 = FILE1 RECORD09 TRUNC
RECORD 4 = FILE1 RECORD11 TRUNC
RECORD 4 = FILE1 RECORD13 TRUNC
RECORD 4 = FILE1 RECORD15 TRUNC
RECORD 4 = FILE1 RECORD17 TRUNC
RECORD 4 = FILE1 RECORD19 TRUNC
RECORD 4 = FILE1 RECORD21 TRUNC
RECORD 4 = FILE1 RECORD23 TRUNC
RECORD 4 = FILE1 RECORD25 TRUNC
RECORD 4 = FILE1 RECORD27 TRUNC
RECORD 4 = FILE1 RECORD29 TRUNC
RECORD 4 = FILE1 RECORD31 TRUNC
RECORD 4 = FILE1 RECORD33 TRUNC
RECORD 4 = FILE1 RECORD35 TRUNC
RECORD 4 = FILE1 RECORD37 TRUNC
RECORD 4 = FILE1 RECORD39 TRUNC
RECORD 4 = FILE1 RECORD41 TRUNC
RECORD 4 = FILE1 RECORD43 TRUNC
RECORD 4 = FILE1 RECORD45 TRUNC
RECORD 4 = FILE1 RECORD47 TRUNC
RECORD 4 = FILE1 RECORD49 TRUNC
CLOSING FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11342724_01.TMP
OPEN FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11342745_02.TMP
MODE = READ
STAT = 00000
AT END OF FILE IISS_SDRG:
[DIR]FS_00002C3D_87320_11342745_02.TMP STATUS
RECORD 4 = FILE2 RECORD19
RECORD 4 = FILE2 RECORD18
RECORD 4 = FILE2 RECORD17
RECORD 4 = FILE2 RECORD16
RECORD 4 = FILE2 RECORD15
RECORD 4 = FILE2 RECORD14
RECORD 4 = FILE2 RECORD13
RECORD 4 = FILE2 RECORD12
RECORD 4 = FILE2 RECORD11
RECORD 4 = FILE2 RECORD10
RECORD 4 = FILE2 RECORD09
RECORD 4 = FILE2 RECORD08
RECORD 4 = FILE2 RECORD07
RECORD 4 = FILE2 RECORD06
RECORD 4 = FILE2 RECORD05
RECORD 4 = FILE2 RECORD04
```


UTP620343400
30 September 1990

RECORD 4 = FILE2 RECORD03
RECORD 4 = FILE2 RECORD02
RECORD 4 = FILE2 RECORD01
CLOSING FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11342745_02.TMP
CLOSING FILE IISS_SDRG:[DIR]FS_00002C3D_87320_11343062_03.TMP
IN = FILET.OUTPUT+IISS_SDRG:[DIR]FS_00002C3D_87320_11342745_02
.TMP+ II
OUT = FILE2T.OUTPUT
FLDS= 1,5,CH,A,14,15,CH,D
SORT STATUS = 00000
SORT COUNT = 000000202
FILENAME = IISS_SDRG:[DIR]FS_00002C3D_87320_11343587_04.TMP
COPY STATUS = 00000
COPY COUNT = 00000202
LAST STATUS = 00000

APPENDIX E

STATUS ERROR CODES

OPNFIL FILE OPEN FUNCTION

00000 - SUCCESS
11101 - INVALID PARAMETER
11102 - FILE OPEN ERROR
11103 - FILE NOT FOUND
11104 - FILE NOT CREATED
11105 - READ ACCESS DENIED
11106 - WRITE ACCESS DENIED
11107 - FILE NOT SEQUENTIAL
11130 - CREATED NEW FILE FOR APPEND

INPFIL FILE READ FUNCTION

00000 - SUCCESS
11201 - INVALID PARAMETER
11202 - FILE NOT OPENED FOR INPUT
11203 - READ ERROR
11210 - END OF FILE
11240 - RECORD TRUNCATED

OUTFIL FILE WRITE FUNCTION

00000 - SUCCESS
11301 - INVALID PARAMETER
11302 - FILE NOT OPENED FOR OUTPUT OR APPEND
11303 - FILE FULL
11304 - WRITE ERROR
11305 - RECORD TOO LARGE, NOT WRITTEN

SEKFIL FILE SEEK FUNCTION

00000 - SUCCESS
11401 - INVALID PARAMETER
11402 - FILE NOT OPENED FOR INPUT
11403 - FILE SEEK ERROR
11410 - END OF FILE
11420 - BEGINNING OF FILE

CLSFIL FILE CLOSE FUNCTION

00000 - SUCCESS
11701 - INVALID PARAMETER
11702 - FILE NOT DELETED
11703 - FILE NOT CLOSED

SRTFIL SORT/MERGE FUNCTION

00000 - SUCCESS
11901 - INVALID PARAMETER
11902 - SORT PACKAGE ERROR
11903 - ERROR CLOSING INPUT FILE(S)
11904 - ERROR CLOSING OUTPUT FILE
11905 - ERROR OPENING INPUT FILE(S)
11906 - ERROR OPENING OUTPUT FILE
11907 - ERROR READING FILE(S)
11908 - INSUFFICIENT WORK SPACE
11909 - ERROR WRITING FILE

APPENDIX F

FILE1T.DAT NOT FOUND ERROR

VAX

11103 OPNFIL File not found
11103 OPNFIL FILE1T.DAT
OPEN FILE FILE1T.DAT
MODE = READ
STAT = 11103
LAST STATUS = 11103

IBM

11103 OPNFIL File not found
11103 OPNFIL USERID.FILE1T.INPUT
OPEN FILE USERID.FILE1T.INPUT
MODE = READ
STAT = 11103
LAST STATUS = 11103

APPENDIX G

FILE2T.DAT NOT FOUND ERROR

VAX

OPEN FILE FILE1T.DAT
MODE = READ
STAT = 00000

11103 OPNFIL File not found
11103 OPNFIL FILE2T.DAT
OPEN FILE FILE2T.DAT
MODE = READ
STAT = 11103
LAST STATUS = 11103

IBM

OPEN FILE USERID.FILE1T.INPUT
MODE = READ
STAT = 00000

11103 OPNFIL File not found
11103 OPNFIL USERID.FILE2T.INPUT
OPEN FILE USERID.FILE2T.INPUT
MODE = READ
STAT = 11103
LAST STATUS = 11103

APPENDIX H

FILE2T.DAT RECORDS NOT FIXED LENGTH ERROR

VAX

OPEN FILE FILE1T.DAT
MODE = READ
STAT = 00000

11102 OPNFIL Records are not Fixed_length
OPEN FILE FILE2T.DAT
MODE = READ
STAT = 11102
LAST STATUS = 11102

IBM

OPEN FILE USERID.FILE1T.INPUT
MODE = READ
STAT = 00000

11102 OPNFIL Records are not Fixed_length
OPEN FILE USERID.FILE2T.INPUT
MODE = READ
STAT = 11102
LAST STATUS = 11102

APPENDIX I

FILE2T OUTPUT FILE 2

FILE2T.OUTPUT ON THE VAX
IISSCM.R23.FILE2TO ON THE IBM

FILE1	RECORD49	TRUNC
FILE1	RECORD49	TRUNC
FILE1	RECORD49	TRUNC
FILE1	RECORD48	TRUNC
FILE1	RECORD48	TRUNC
FILE1	RECORD47	TRUNC
FILE1	RECORD47	TRUNC
FILE1	RECORD47	TRUNC
FILE1	RECORD46	TRUNC
FILE1	RECORD46	TRUNC
FILE1	RECORD45	TRUNC
FILE1	RECORD45	TRUNC
FILE1	RECORD45	TRUNC
FILE1	RECORD44	TRUNC
FILE1	RECORD44	TRUNC
FILE1	RECORD43	TRUNC
FILE1	RECORD43	TRUNC
FILE1	RECORD43	TRUNC
FILE1	RECORD42	TRUNC
FILE1	RECORD42	TRUNC
FILE1	RECORD41	TRUNC
FILE1	RECORD41	TRUNC
FILE1	RECORD41	TRUNC
FILE1	RECORD40	TRUNC
FILE1	RECORD40	TRUNC
FILE1	RECORD39	TRUNC
FILE1	RECORD39	TRUNC
FILE1	RECORD39	TRUNC
FILE1	RECORD38	TRUNC
FILE1	RECORD38	TRUNC
FILE1	RECORD37	TRUNC
FILE1	RECORD37	TRUNC
FILE1	RECORD37	TRUNC
FILE1	RECORD36	TRUNC
FILE1	RECORD36	TRUNC
FILE1	RECORD35	TRUNC
FILE1	RECORD35	TRUNC
FILE1	RECORD35	TRUNC
FILE1	RECORD34	TRUNC
FILE1	RECORD34	TRUNC
FILE1	RECORD33	TRUNC
FILE1	RECORD33	TRUNC
FILE1	RECORD33	TRUNC
FILE1	RECORD33	TRUNC
FILE1	RECORD32	TRUNC
FILE1	RECORD32	TRUNC
FILE1	RECORD31	TRUNC
FILE1	RECORD31	TRUNC
FILE1	RECORD31	TRUNC

FILE1	RECORD30	TRUNC
FILE1	RECORD30	TRUNC
FILE1	RECORD29	TRUNC
FILE1	RECORD29	TRUNC
FILE1	RECORD29	TRUNC
FILE1	RECORD28	TRUNC
FILE1	RECORD28	TRUNC
FILE1	RECORD27	TRUNC
FILE1	RECORD27	TRUNC
FILE1	RECORD27	TRUNC
FILE1	RECORD26	TRUNC
FILE1	RECORD26	TRUNC
FILE1	RECORD25	TRUNC
FILE1	RECORD25	TRUNC
FILE1	RECORD25	TRUNC
FILE1	RECORD24	TRUNC
FILE1	RECORD24	TRUNC
FILE1	RECORD23	TRUNC
FILE1	RECORD23	TRUNC
FILE1	RECORD23	TRUNC
FILE1	RECORD22	TRUNC
FILE1	RECORD22	TRUNC
FILE1	RECORD21	TRUNC
FILE1	RECORD21	TRUNC
FILE1	RECORD21	TRUNC
FILE1	RECORD20	TRUNC
FILE1	RECORD20	TRUNC
FILE1	RECORD19	TRUNC
FILE1	RECORD19	TRUNC
FILE1	RECORD19	TRUNC
FILE1	RECORD18	TRUNC
FILE1	RECORD18	TRUNC
FILE1	RECORD17	TRUNC
FILE1	RECORD17	TRUNC
FILE1	RECORD17	TRUNC
FILE1	RECORD16	TRUNC
FILE1	RECORD16	TRUNC
FILE1	RECORD15	TRUNC
FILE1	RECORD15	TRUNC
FILE1	RECORD15	TRUNC
FILE1	RECORD14	TRUNC
FILE1	RECORD14	TRUNC
FILE1	RECORD13	TRUNC
FILE1	RECORD13	TRUNC
FILE1	RECORD13	TRUNC
FILE1	RECORD12	TRUNC
FILE1	RECORD12	TRUNC
FILE1	RECORD11	TRUNC
FILE1	RECORD11	TRUNC
FILE1	RECORD11	TRUNC
FILE1	RECORD10	TRUNC
FILE1	RECORD10	TRUNC
FILE1	RECORD09	TRUNC
FILE1	RECORD09	TRUNC
FILE1	RECORD09	TRUNC
FILE1	RECORD08	TRUNC
FILE1	RECORD08	TRUNC

FILE1	RECORD07	TRUNC
FILE1	RECORD07	TRUNC
FILE1	RECORD07	TRUNC
FILE1	RECORD06	TRUNC
FILE1	RECORD06	TRUNC
FILE1	RECORD05	TRUNC
FILE1	RECORD05	TRUNC
FILE1	RECORD05	TRUNC
FILE1	RECORD04	TRUNC
FILE1	RECORD04	TRUNC
FILE1	RECORD03	TRUNC
FILE1	RECORD03	TRUNC
FILE1	RECORD03	TRUNC
FILE1	RECORD02	TRUNC
FILE1	RECORD02	TRUNC
FILE1	RECORD01	TRUNC
FILE1	RECORD01	TRUNC
FILE1	RECORD01	TRUNC
FILE2	RECORD20	TRUNC
FILE2	RECORD20	
FILE2	RECORD20	
FILE2	RECORD19	
FILE2	RECORD19	
FILE2	RECORD19	
FILE2	RECORD19	
FILE2	RECORD18	
FILE2	RECORD18	
FILE2	RECORD18	
FILE2	RECORD18	
FILE2	RECORD17	
FILE2	RECORD17	
FILE2	RECORD17	
FILE2	RECORD17	
FILE2	RECORD16	
FILE2	RECORD16	
FILE2	RECORD16	
FILE2	RECORD16	
FILE2	RECORD15	
FILE2	RECORD15	
FILE2	RECORD15	
FILE2	RECORD15	
FILE2	RECORD14	
FILE2	RECORD14	
FILE2	RECORD14	
FILE2	RECORD14	
FILE2	RECORD13	
FILE2	RECORD13	
FILE2	RECORD13	
FILE2	RECORD13	
FILE2	RECORD12	
FILE2	RECORD12	
FILE2	RECORD12	
FILE2	RECORD12	
FILE2	RECORD11	
FILE2	RECORD11	
FILE2	RECORD11	
FILE2	RECORD11	

FILE2 RECORD10
FILE2 RECORD10
FILE2 RECORD10
FILE2 RECORD10
FILE2 RECORD09
FILE2 RECORD09
FILE2 RECORD09
FILE2 RECORD09
FILE2 RECORD08
FILE2 RECORD08
FILE2 RECORD08
FILE2 RECORD08
FILE2 RECORD08
FILE2 RECORD07
FILE2 RECORD07
FILE2 RECORD07
FILE2 RECORD07
FILE2 RECORD06
FILE2 RECORD06
FILE2 RECORD06
FILE2 RECORD06
FILE2 RECORD06
FILE2 RECORD05
FILE2 RECORD05
FILE2 RECORD05
FILE2 RECORD05
FILE2 RECORD04
FILE2 RECORD04
FILE2 RECORD04
FILE2 RECORD04
FILE2 RECORD03
FILE2 RECORD03
FILE2 RECORD03
FILE2 RECORD03
FILE2 RECORD02
FILE2 RECORD02
FILE2 RECORD02
FILE2 RECORD02
FILE2 RECORD01
FILE2 RECORD01
FILE2 RECORD01
FILE2 RECORD01